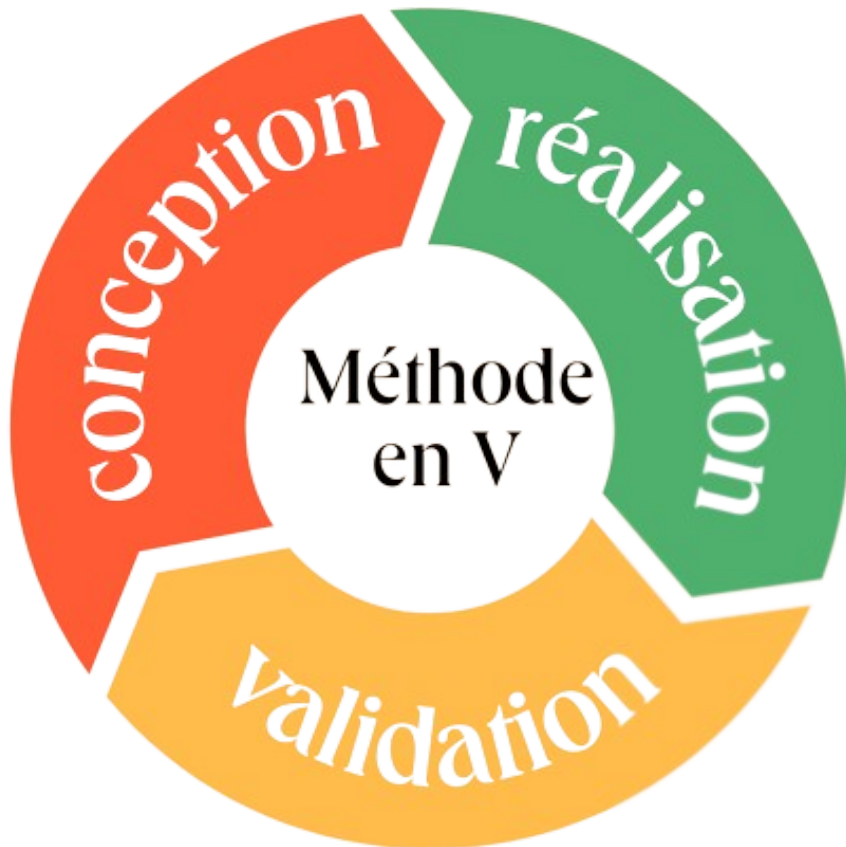


GESTION DE PROJET

Cycle en V



GESTION DE PROJET

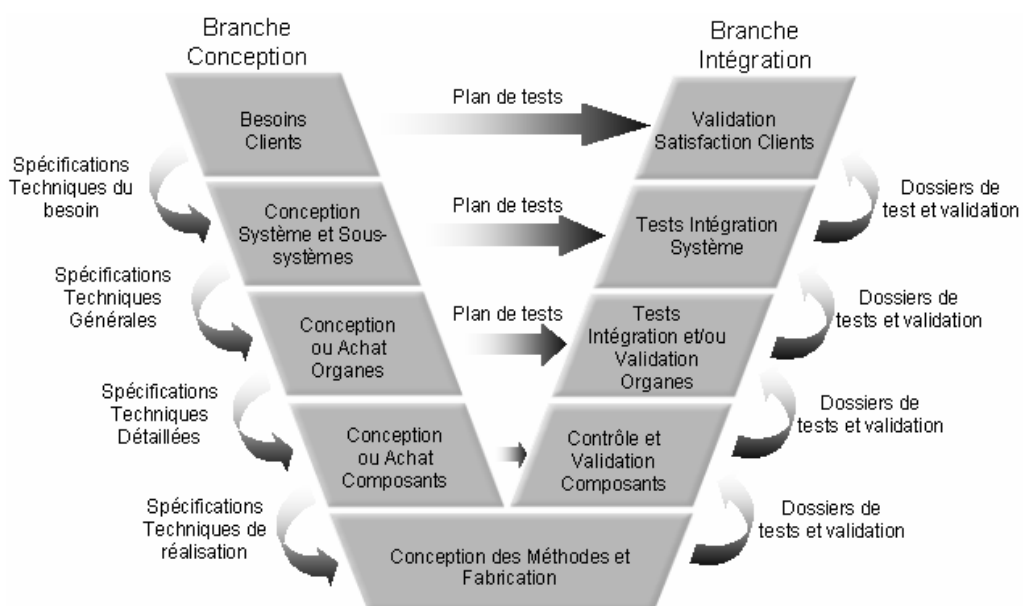
Cycle en V

Introduction

Le cycle en V en gestion de projet découle du modèle en cascade théorisé dans les années 1970, qui permet de représenter des processus de développement de manière linéaire et en phases successives.

Ce mode de gestion de projet a été **développé dans les années 1980** et appliqué au champ des projets industriels, puis étendu aux projets informatiques. Il a été remis en cause à partir du début des années 2000, sous l'effet de l'accélération des changements technologiques, favorisant davantage les méthodes dites « agiles ».

La lettre V fait référence à la vision schématique de ce cycle, qui prend la forme d'un V : une phase descendante suivie d'une phase ascendante. Le cycle en V associe à chaque phase de réalisation une phase de validation, comme l'illustre le schéma ci-dessous :



Qu'est-ce que la méthode cycle en V ?

Le **cycle en V** est un modèle de gestion de projet qui implique toutes les étapes du cycle de vie d'un projet : conception, réalisation et validation. Cette méthode comporte une phase descendante, puis une phase ascendante, illustrées par les deux branches du V.

Issue du **waterfall model**, ou modèle de gestion de projet en cascade, la **méthode du cycle en V** repose sur des **étapes séquentielles et linéaires**, allant de l'analyse des **besoins** au **test d'acceptation**. La partie descendante du « V » correspond aux quatre actions de conception et de développement du système, tandis que la partie ascendante reprend les quatre phases d'assurance qualité qui lui sont associées. En effet, la particularité de ce modèle de gestion de projet est de combiner une phase de validation pour chaque phase de développement. Le point de jonction, le bas du V, correspond quant à elle à l'étape de réalisation.

1-Les principales caractéristiques de la méthode cycle en V

La méthode du cycle en V implique toutes les étapes du cycle de mise en œuvre d'un produit ou d'un logiciel. Elle permet de définir le processus d'un projet en neuf étapes regroupées en trois phases : la conception, la réalisation et la validation.

La conception

Au départ, la phase de conception joue un rôle essentiel en rassemblant les besoins et les spécifications essentiels pour orienter avec précision tout le projet. Elle implique également la compilation préalable des éléments indispensables à la réalisation du projet. Dans le cadre de la méthodologie du Cycle en V pour le développement de logiciels, cette phase de conception comprend quatre étapes distinctes : l'analyse des besoins, la définition des spécifications, la conception générale et la conception détaillée.

En première étape, l'analyse des besoins se concentre sur la compréhension des exigences de l'entreprise. Elle vise à saisir les attentes et les demandes du client en posant des questions essentielles sur divers critères, tels que le budget et le résultat final escompté. Ces besoins servent de base pour identifier les utilisations prévues du produit final. Dans certains cas, l'analyse des besoins peut également inclure une évaluation de la faisabilité du projet.

La deuxième étape implique la définition des spécifications fonctionnelles par le biais de la création d'un cahier des charges fonctionnel. Ce document synthétise les exigences précises du

client en indiquant ce qui doit être réalisé, sans entrer dans les détails des méthodes pour y parvenir.

La troisième étape, connue sous le nom de conception générale ou conception architecturale, transforme les spécifications fonctionnelles en spécifications techniques. L'équipe de projet passe en revue les technologies essentielles nécessaires à la mise en œuvre du logiciel et élabore une architecture répondant aux besoins du client. Cette phase peut également entraîner des décisions techniques et financières liées à la viabilité du projet. En cas d'obstacles techniques, le chef de projet peut réexaminer certaines fonctionnalités et apporter des modifications au cahier des charges.

Enfin, la quatrième étape de la conception détaillée vise à préciser tous les composants indispensables à la création du produit ou du logiciel. Elle détermine également la manière dont ces composants interagiront pour répondre aux besoins du client. Cette étape se matérialise par la rédaction d'un document qui peut contenir, par exemple, des instructions de codage nécessaires au développement logiciel.

La réalisation

Après la conception vient l'étape de réalisation qui permet de créer et d'assembler tous les composants nécessaires pour arriver au produit final. Il peut s'agir de la fabrication d'un équipement ou du codage d'un logiciel. Dans ce dernier cas, l'équipe choisit le langage de programmation en fonction des attentes fonctionnelles et architecturales établies lors de la phase de conception. L'étape de réalisation doit ensuite suivre les directives et normes de codage en vigueur.

La validation

La phase de validation comprend différents tests vérifiant chaque étape de la conception du produit. Ces mesures d'assurance qualité prennent la forme de tests unitaires, de tests d'intégration, de tests de validation et d'une recette fonctionnelle.

Les tests unitaires correspondent à l'étape de conception détaillée. Réalisés au niveau du code par le développeur, ils permettent de détecter et de corriger d'éventuelles erreurs. L'équipe contrôle ainsi le bon fonctionnement de chaque brique logicielle.

Les tests d'intégration interviennent sur le produit fini et vont permettre de veiller au respect du cahier des charges technique édité lors de la phase de conception architecturale. Regroupant des essais techniques et des essais fonctionnels, les tests d'intégration visent à s'assurer que le système fonctionne dans son ensemble.

Les tests de validation correspondent à la phase des spécifications. Menés par les futurs utilisateurs du logiciel, les tests de validation sont basés sur des mises en situation réelle pour vérifier la conformité fonctionnelle du logiciel par rapport aux spécifications souhaitées par le client.

À la fin du cycle de vie du projet, la recette fonctionnelle, appelée également test d'acceptation, permet au client de valider le produit par rapport aux besoins et aux exigences exprimés lors de la première phase, celle de l'analyse de besoins. La recette fonctionnelle est la dernière séquence avant la mise en production du logiciel.

2-Quelles sont les 9 étapes de la méthode cycle en V ?

1 – Analyse des besoins

Lors de cette étape, **les besoins du client ou futur utilisateur sont précis**. Ceci dans le but de **définir les fonctionnalités et les demandes du client**. Il est crucial d'allouer assez de temps à cette étape et de **réunir toutes les exigences du client**. Comme pour la phase d'analyse des besoins du modèle en cascade, le bon déroulé du projet en dépend.

Dans le **cycle en V**, à chaque phase de conception, **les tests unitaires sont créés**. Les tests unitaires d'acceptation sont donc à créer lors de cette première étape.

2 – Les spécifications

Au cours de cette étape, un **document contenant les spécifications fonctionnelles du produit est créé**. Il contient **tous les composants techniques** en se basant sur la définition des besoins réalisés lors de l'étape précédente.

Pendant cette étape, des **tests unitaires du système sont également mis au point**. Pour une mise en œuvre ultérieure.

A voir aussi : **Cahier des charges : comment le réaliser ?**

3 – La conception de l'architecture

C'est le moment où **les spécifications fonctionnelles sur l'intégration du programme sont rédigées**. Il faut **spécifier si le programme connecte ses composants via une intégration interne ou externe**. Cette phase est aussi appelée **conception de haut niveau**.

Pendant cette étape, **des tests d'intégration sont également créés**.

4 – La conception détaillée

Ici on parle de **phase de conception de bas niveau du système**. Elle concerne la façon dont sera mise en œuvre la logique fonctionnelle codée du produit final. Notamment, **les spécifications sur les composants, les modèles et les interfaces**.

5 – Le codage

Nous sommes à mi-chemin du cycle en V. C'est à ce moment qu'ont lieu **la mise en œuvre et le véritable codage**. Tous les documents de spécifications et de conception créés précédemment doivent être **transformés en un système codé et fonctionnel**.

Cette étape doit être clôturée avant que la phase de **tests débute**.

6 – Les tests unitaires

C'est la **première étape de la phase ascendante du V**. Les tests unitaires créés pendant la phase de conception des modules sont exécutés. Cela permet d'identifier et éliminer une grande partie des défauts du produit. Ils représentent normalement l'étape la plus longue dans le **cycle en V** pour la gestion de projet informatique.

Les tests unitaires **ne permettent généralement pas d'identifier tous les défauts qui pourraient survenir dans le système**. C'est pourquoi il y a d'autres étapes, comme les tests d'intégration, qui permettent de découvrir d'autres lacunes éventuelles.

7 – Les tests d'intégration

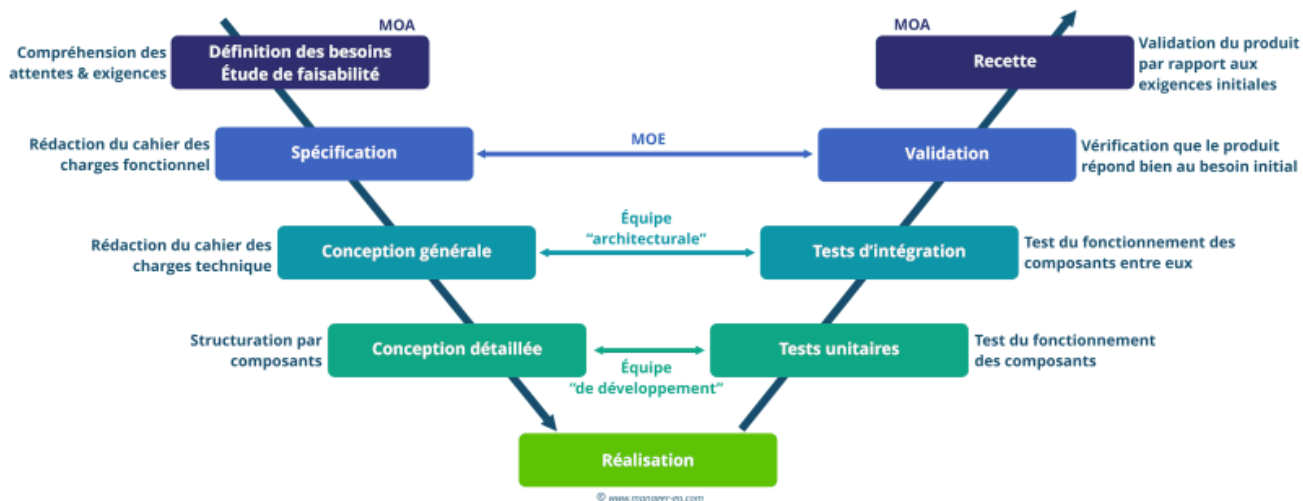
Ils servent à **vérifier que le système fonctionne sur toutes les intégrations tierces**. Notamment concernant les composants. Ces tests d'intégration se basent sur les résultats prévus pendant la phase de conception architecturale.

8 – Les tests de validation

C'est le moment de la **réalisation des tests de validation**, créés pendant l'étape de **conception du système**. Ils se composent plus particulièrement de tests de performance et de régression.

9 – La recette ou vérification

Dernière étape du cycle en V pour la gestion de projet informatique. C'est **la mise en œuvre de tous les tests créés lors du stade initial de définition des exigences**. La **réalisation de ces tests se fait dans un environnement réel, avec des données réelles**. Ceci a pour but de vérifier que le **produit est prêt à être livré au client**.



3-Les Rôles (Qui valide ?)

Dans le modèle du Cycle en V, différentes personnes ou entités sont responsables de la validation à chaque étape du processus. Voici un aperçu des rôles et des responsabilités par niveau de détail, conformément aux séquences du Cycle en V :

Maîtrise d'ouvrage (fonctionnel), ou MOA :

Rôle : La maîtrise d'ouvrage représente le client ou les parties prenantes qui définissent les besoins et les attentes du projet. Elle est responsable de la validation à l'étape de l'analyse des besoins et des spécifications.

Responsabilités : La MOA doit s'assurer que les exigences sont correctement compréhensibles et qu'elles sont bien documentées. Elle valide que les spécifications fonctionnelles reflètent correctement les besoins.

Maîtrise d'œuvre (système), ou MOE :

Rôle : La maîtrise d'œuvre est l'entité responsable de la gestion et de la mise en œuvre du projet. Elle supervise l'ensemble du processus de développement.

Responsabilités : La MOE est impliquée dans toutes les phases du Cycle en V, mais elle joue un rôle majeur dans la conception générale, la conception détaillée, le développement, les tests d'intégration, les tests de validation et la vérification. Elle s'assure que le projet progresse conformément aux spécifications et aux exigences définies par la MOA.

L'équipe architecturale, ou de conception générale (technique, métier) :

Rôle : L'équipe architecturale est responsable de la conception technique du système. Elle traduit les spécifications fonctionnelles en spécifications techniques et élabore l'architecture du système.

Responsabilités : Cette équipe est particulièrement active dans les phases de conception générale et de conception détaillée. Elle conçoit l'architecture, établit les normes techniques et guide l'équipe de développement dans la mise en œuvre.

L'équipe de développement (par composant) :

Rôle : L'équipe de développement est chargée de créer le code source et de développer les composants du système.

Responsabilités : Elle intervient principalement dans les phases de développement et de tests unitaires. Chaque membre de l'équipe est responsable de la création, du codage, et des tests des composants spécifiques qui lui sont assignés.

Chacun de ces rôles et responsabilités est essentiel pour garantir que le système est correctement conçu, développé et validé tout au long du processus du Cycle en V. La clarté des rôles et des responsabilités est cruciale pour assurer la cohérence et la qualité du produit final.

4-Les Avantages de la méthodologie du Cycle V

Le modèle du Cycle en V présente plusieurs avantages significatifs dans le développement de projets, notamment :

Cohérence entre les étapes de développement et de validation : Le Cycle en V établit une correspondance étroite entre chaque étape de développement et une étape de validation correspondante. Cette cohérence garantit que le produit final est en adéquation avec les exigences et les spécifications tout au long du processus, réduisant ainsi le risque d'erreurs coûteuses.

Validation précoce : La méthodologie du Cycle en V encourage la validation précoce du système. Les tests et les vérifications commencent dès les premières étapes du projet, ce qui permet de détecter et de corriger les erreurs à un stade précoce. Cela réduit les coûts de correction et améliore la qualité globale du produit.

Gestion efficace des risques : Le modèle intègre une gestion proactive des risques tout au long du projet. Les risques potentiels sont identifiés dès le début, évalués et gérés de manière appropriée. Cela permet de minimiser les surprises et les retards liés aux problèmes imprévus.

Documentation claire : Chaque phase du Cycle en V génère des livrables spécifiques, tels que des spécifications des exigences, des schémas architecturaux, des conceptions détaillées, des rapports de tests, etc. Cette documentation claire facilite la communication, la traçabilité des décisions et la gestion du projet.

Amélioration de la qualité : En mettant l'accent sur la validation précoce, la gestion des risques et la cohérence entre les étapes, le Cycle en V vise à améliorer la qualité globale du produit final. En identifiant et en corrigeant les erreurs plus tôt dans le processus, il contribue à fournir un produit plus fiable et conforme aux exigences.

Visibilité du progrès : Le modèle du Cycle en V offre une visibilité claire du progrès du projet. À chaque étape, il est possible d'évaluer l'état d'avancement en fonction des livrables produits, ce qui facilite la gestion et la planification.

Adaptabilité : Bien que le Cycle en V suive une séquence linéaire, il peut être adapté en fonction des besoins spécifiques du projet. Par exemple, certaines phases peuvent être itérées si nécessaire pour résoudre des problèmes identifiés.

Minimisation des retours en arrière : En détectant les problèmes tôt dans le processus, le Cycle en V minimise la nécessité de retours en arrière coûteux et de modifications majeures une fois le développement avancé.

Satisfaction du client : En fournissant un produit de haute qualité qui répond aux exigences du client, le Cycle en V favorise la satisfaction du client et renforce la confiance dans le processus de développement.

En résumé, le modèle du Cycle en V est conçu pour assurer la cohérence, la qualité et la gestion des risques tout au long du processus de développement, ce qui en fait une méthodologie attrayante pour de nombreux projets, en particulier ceux où la fiabilité et la conformité aux exigences sont essentielles.

Conclusion

Le modèle du Cycle en V est une méthodologie de développement de logiciels qui présente plusieurs avantages clés. Il assure la cohérence entre les étapes de développement et de validation, encourage la validation précoce, permet une gestion efficace des risques et génère une documentation claire à chaque étape du projet. En mettant l'accent sur la qualité et la satisfaction du client, le Cycle en V favorise la minimisation des erreurs, la visibilité du progrès du projet et la réduction des coûts liés à la correction tardive des problèmes. Il s'agit d'une méthodologie adaptable qui peut être ajustée en fonction des besoins spécifiques du projet, ce qui en fait un choix efficace pour de nombreux projets de développement de logiciels.